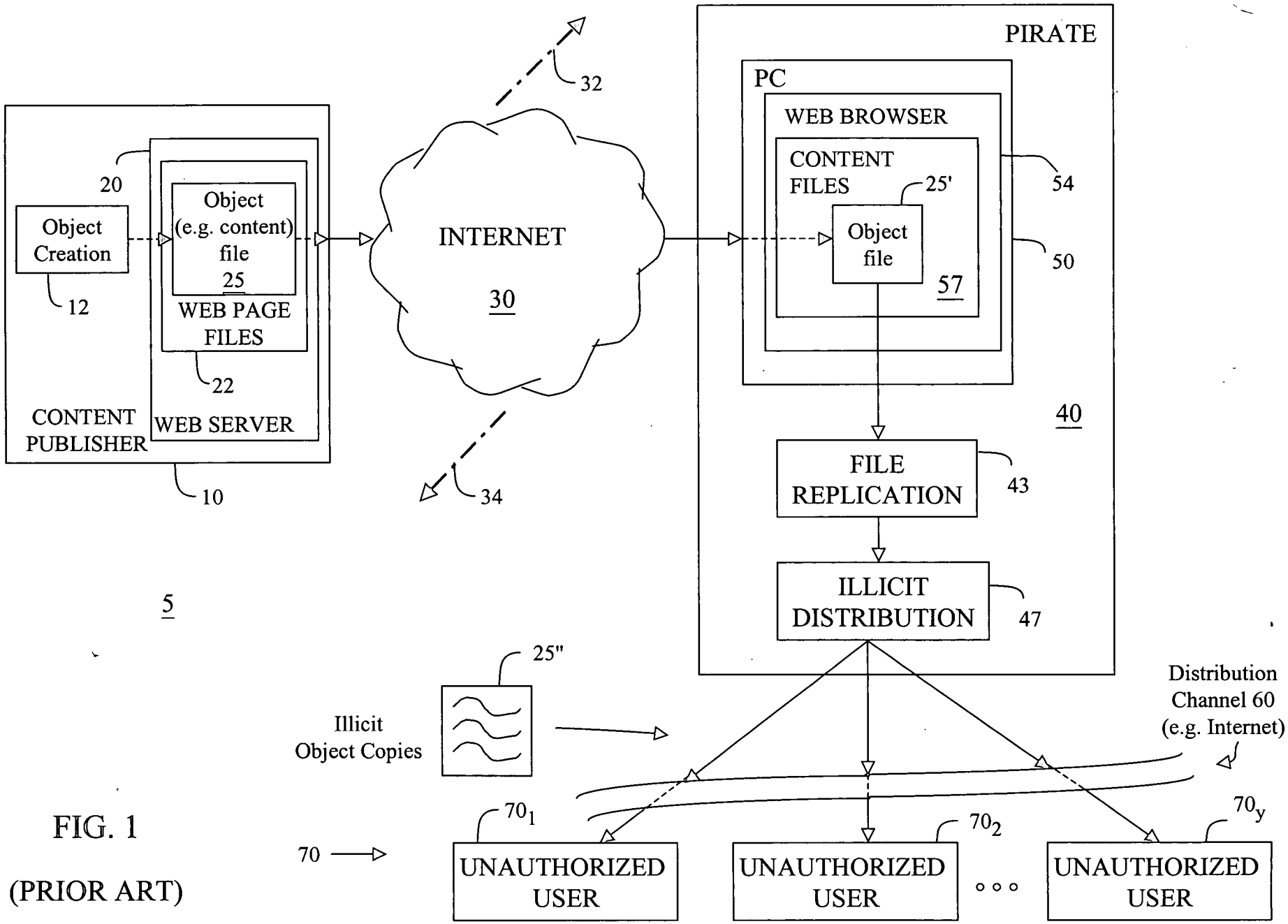


FIG. 1

(PRIOR ART)

FIG. 2 (PRIOR ART)

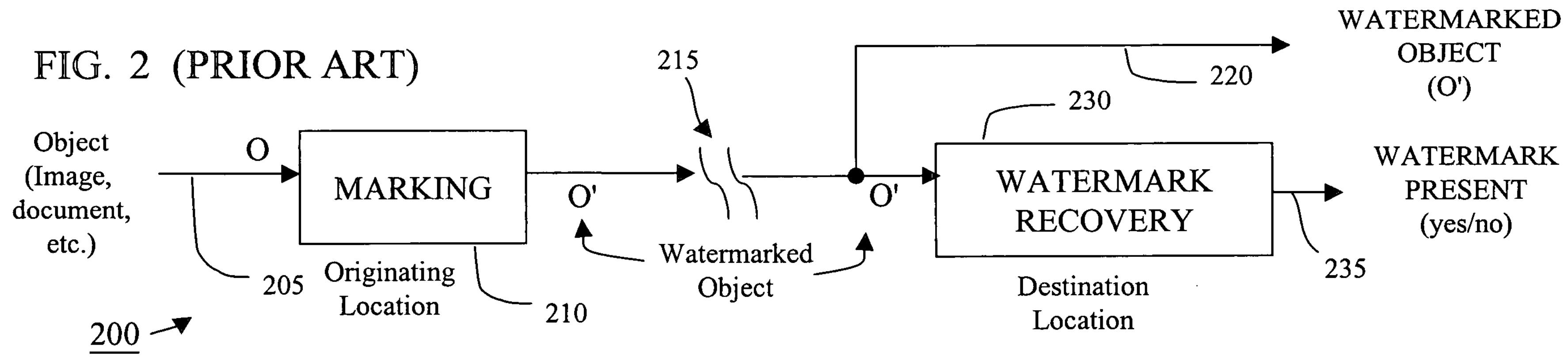


FIG. 3

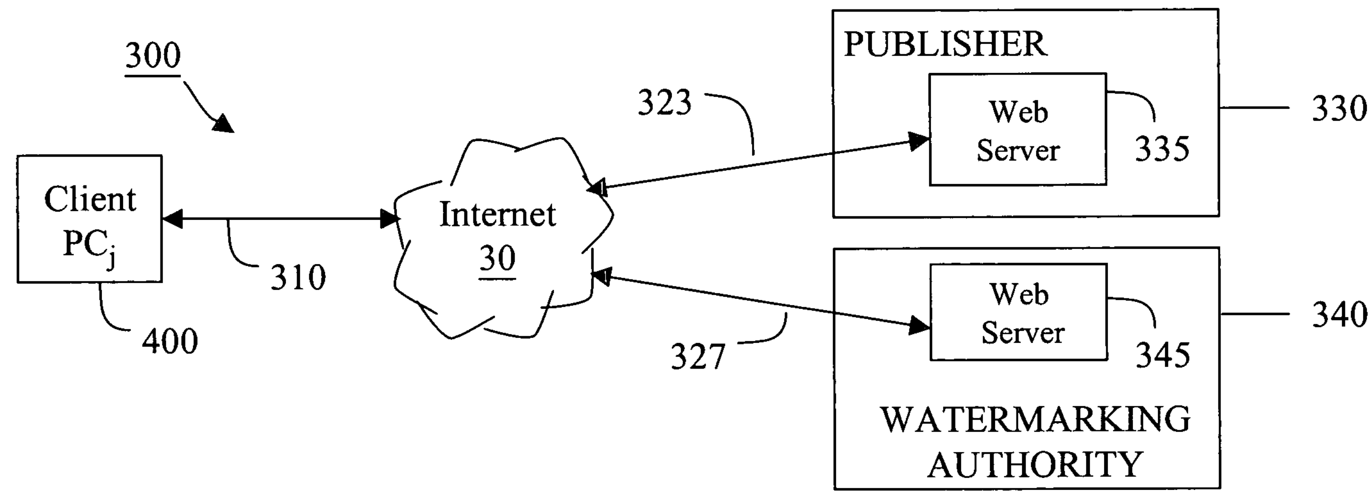


FIG. 10

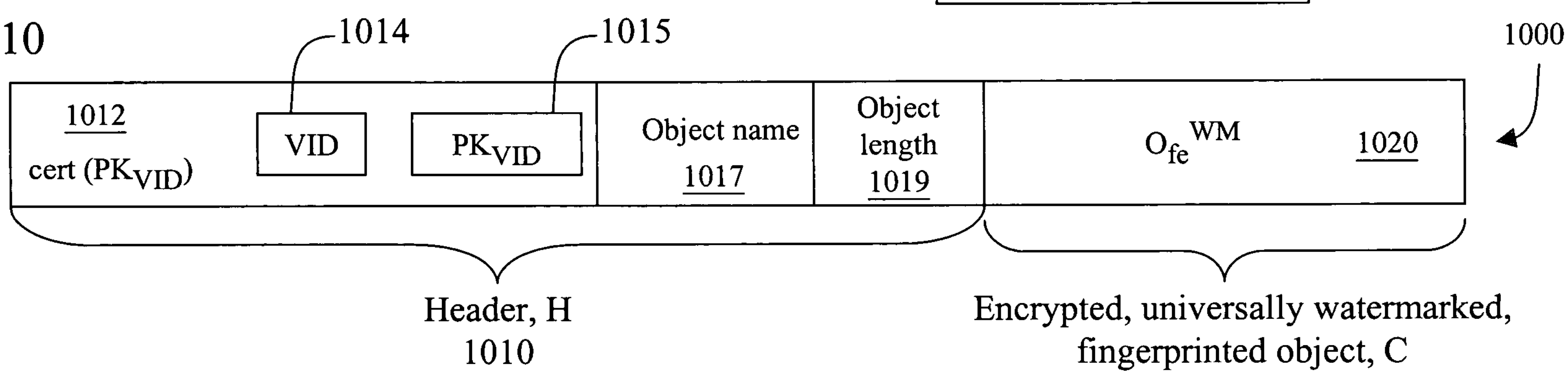
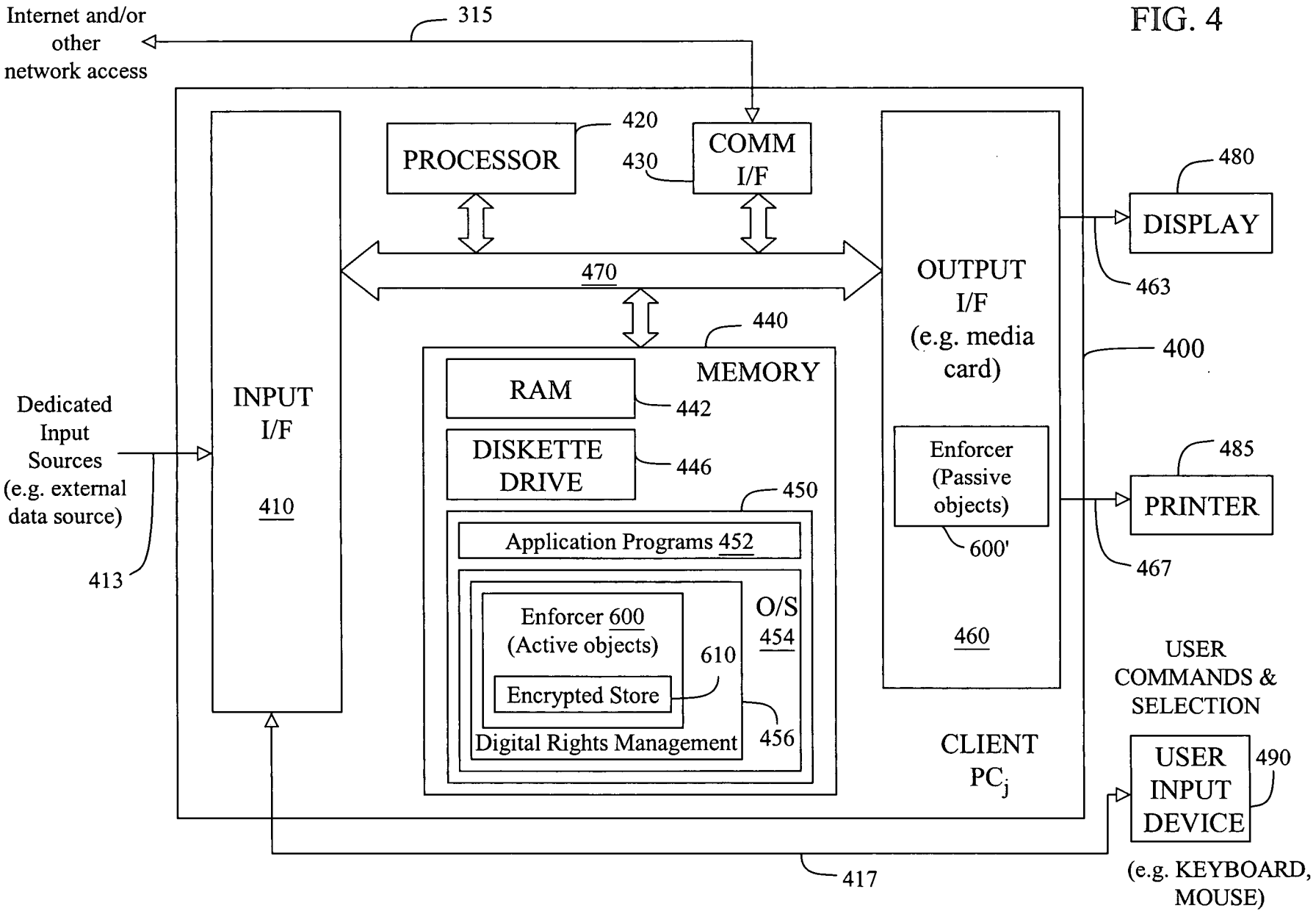


FIG. 4


Internet and/or other network access
315
PROCESSOR
420
COMM I/F
430
470
INPUT I/F
410
Dedicated Input Sources (e.g. external data source)
413
440
MEMORY
RAM
442
DISKETTE DRIVE
446
450
Application Programs 452
Enforcer 600 (Active objects)
610
Encrypted Store
Digital Rights Management
456
O/S
454
OUTPUT I/F (e.g. media card)
460
Enforcer (Passive objects)
600'
CLIENT PC$_j$
400
480
DISPLAY
463
485
PRINTER
467
USER COMMANDS & SELECTION
USER INPUT DEVICE
490
(e.g. KEYBOARD, MOUSE)
417

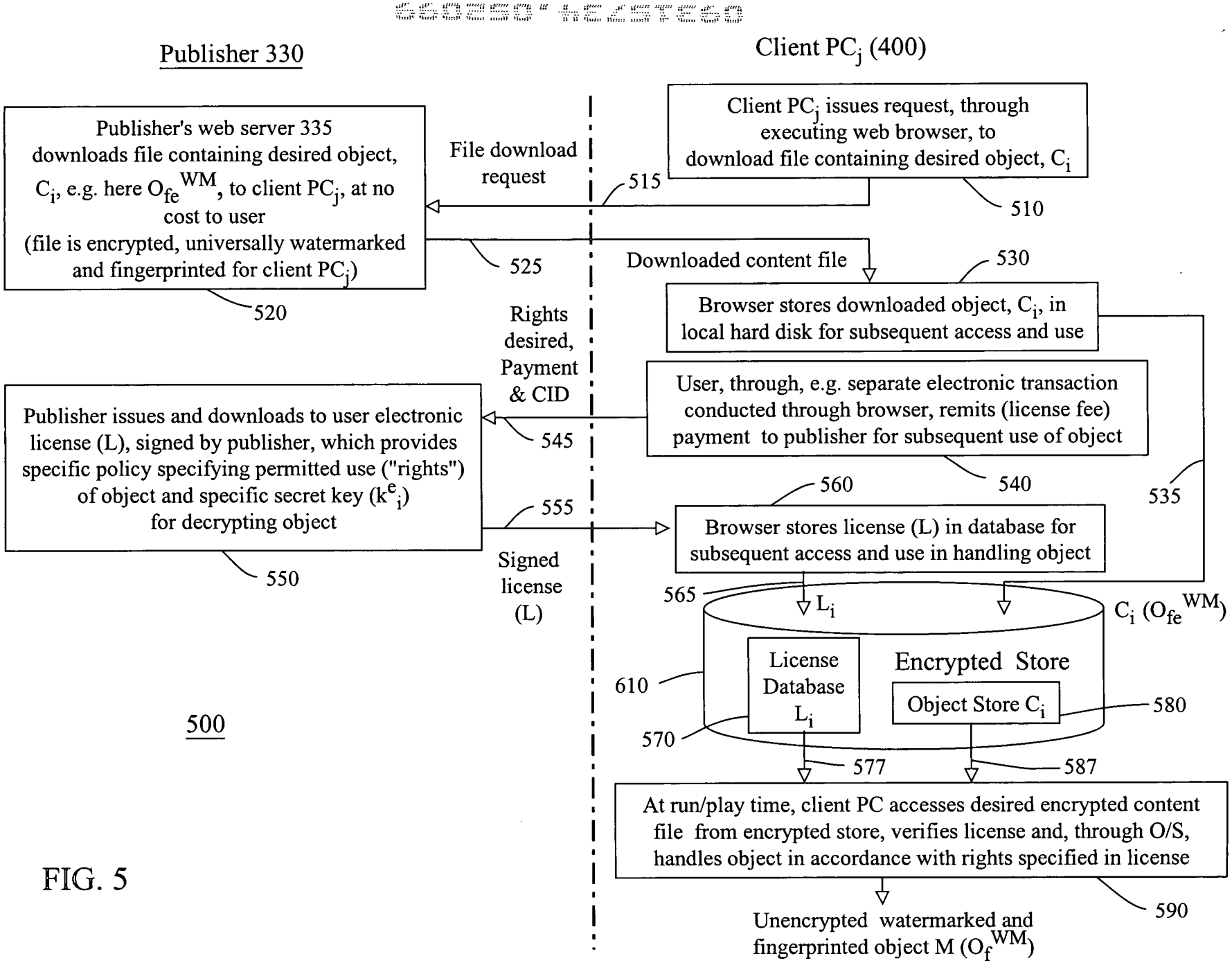

Publisher 330
Client PC$_j$ (400)
Publisher's web server 335 downloads file containing desired object, C_i, e.g. here O_{fe}^{WM}, to client PC$_j$, at no cost to user (file is encrypted, universally watermarked and fingerprinted for client PC$_j$)
520
File download request
515
Client PC$_j$ issues request, through executing web browser, to download file containing desired object, C_i
510
525
Downloaded content file
530
Browser stores downloaded object, C_i, in local hard disk for subsequent access and use
Rights desired, Payment & CID
User, through, e.g. separate electronic transaction conducted through browser, remits (license fee) payment to publisher for subsequent use of object
540
535
Publisher issues and downloads to user electronic license (L), signed by publisher, which provides specific policy specifying permitted use ("rights") of object and specific secret key (k^e_i) for decrypting object
545
555
550
Signed license (L)
560
Browser stores license (L) in database for subsequent access and use in handling object
565
L_i
C_i (O_{fe}^{WM})
License Database L_i
Encrypted Store
Object Store C_i
610
570
580
577
587
At run/play time, client PC accesses desired encrypted content file from encrypted store, verifies license and, through O/S, handles object in accordance with rights specified in license
590
Unencrypted watermarked and fingerprinted object M (O_f^{WM})
500

FIG. 5

FIG. 6

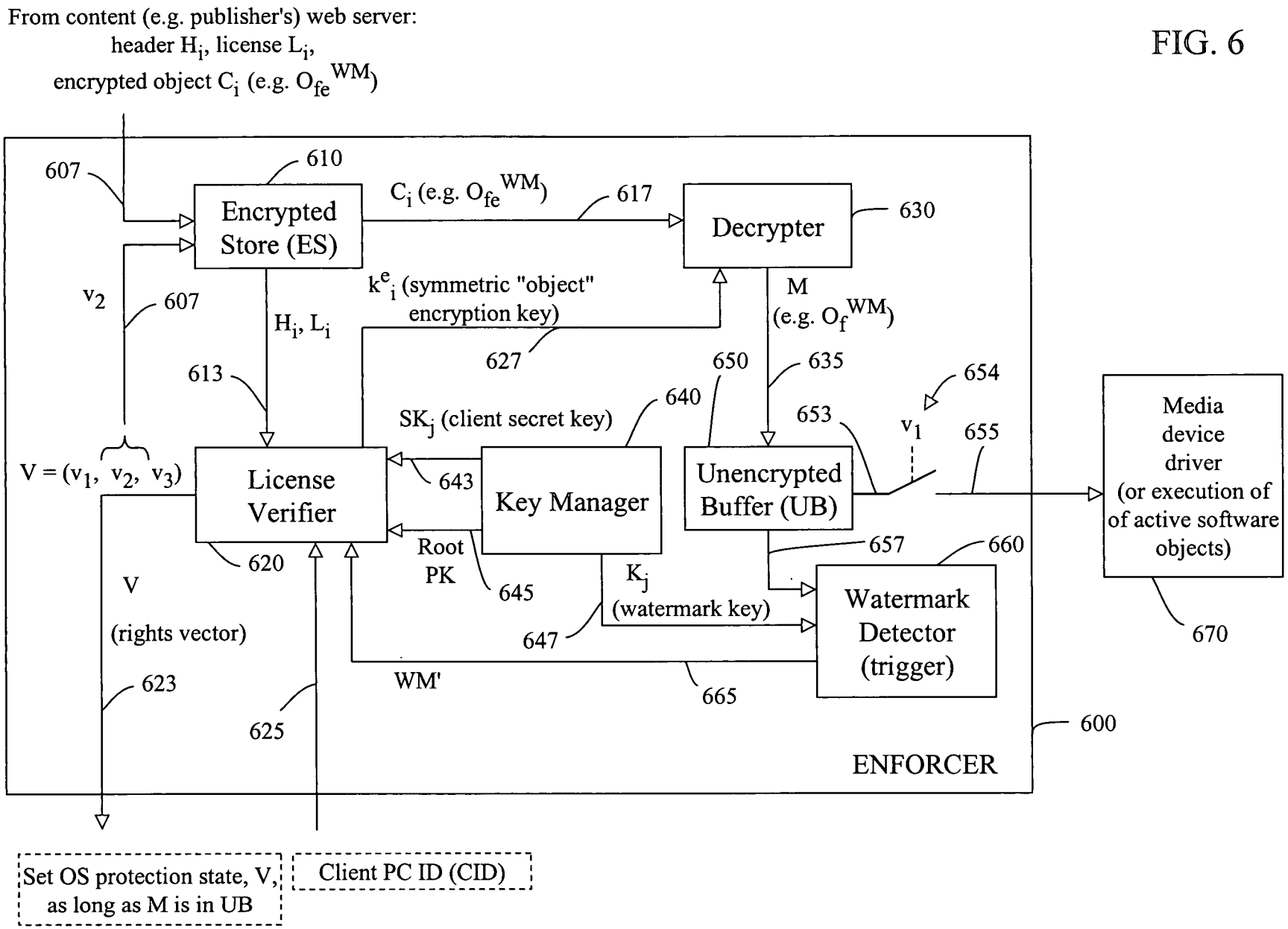
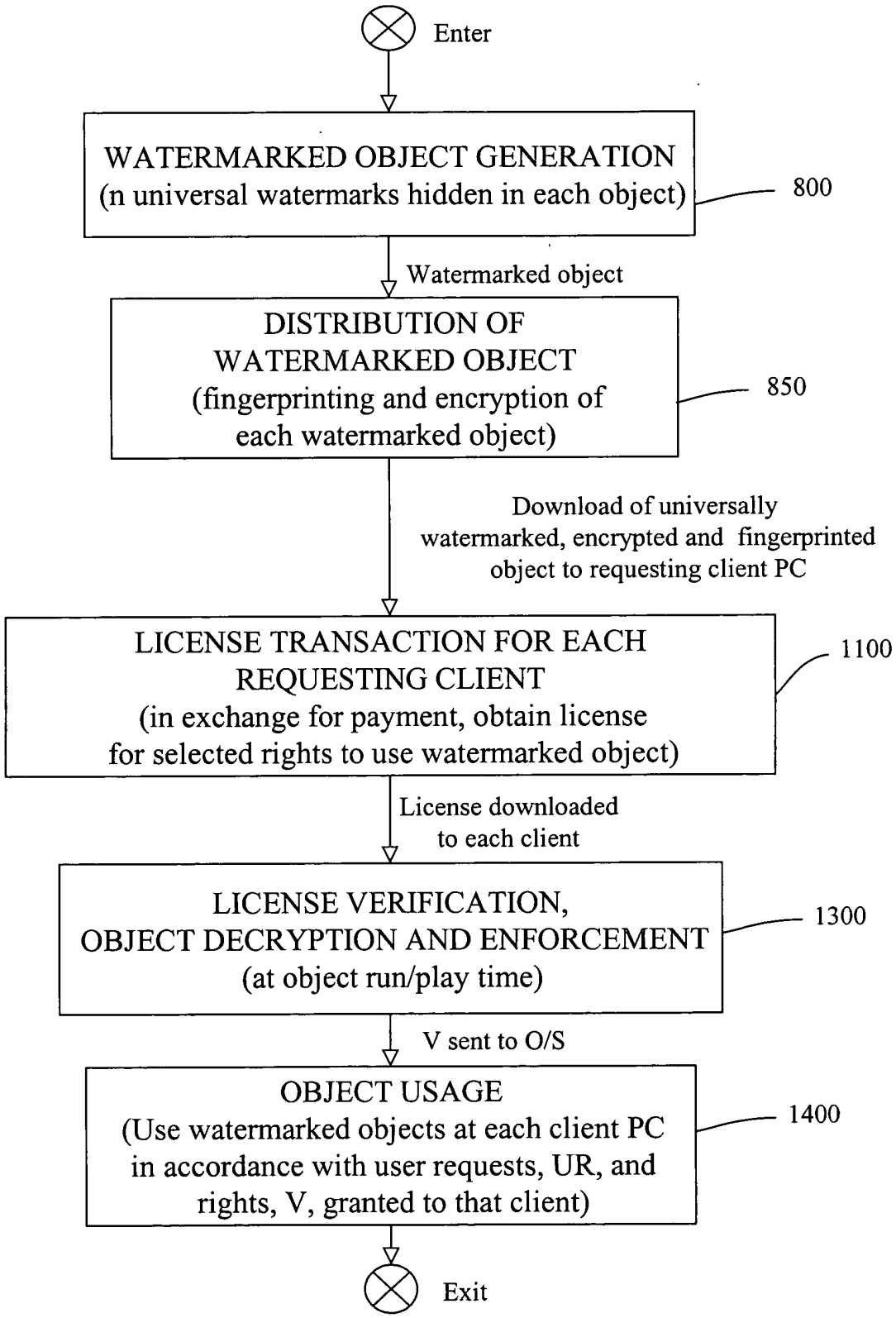


FIG. 7


700
Enter
WATERMARKED OBJECT GENERATION
(n universal watermarks hidden in each object)
800
Watermarked object
DISTRIBUTION OF
WATERMARKED OBJECT
(fingerprinting and encryption of
each watermarked object)
850
Download of universally
watermarked, encrypted and fingerprinted
object to requesting client PC
LICENSE TRANSACTION FOR EACH
REQUESTING CLIENT
(in exchange for payment, obtain license
for selected rights to use watermarked object)
1100
License downloaded
to each client
LICENSE VERIFICATION,
OBJECT DECRYPTION AND ENFORCEMENT
(at object run/play time)
1300
V sent to O/S
OBJECT USAGE
(Use watermarked objects at each client PC
in accordance with user requests, UR, and
rights, V, granted to that client)
1400
Exit

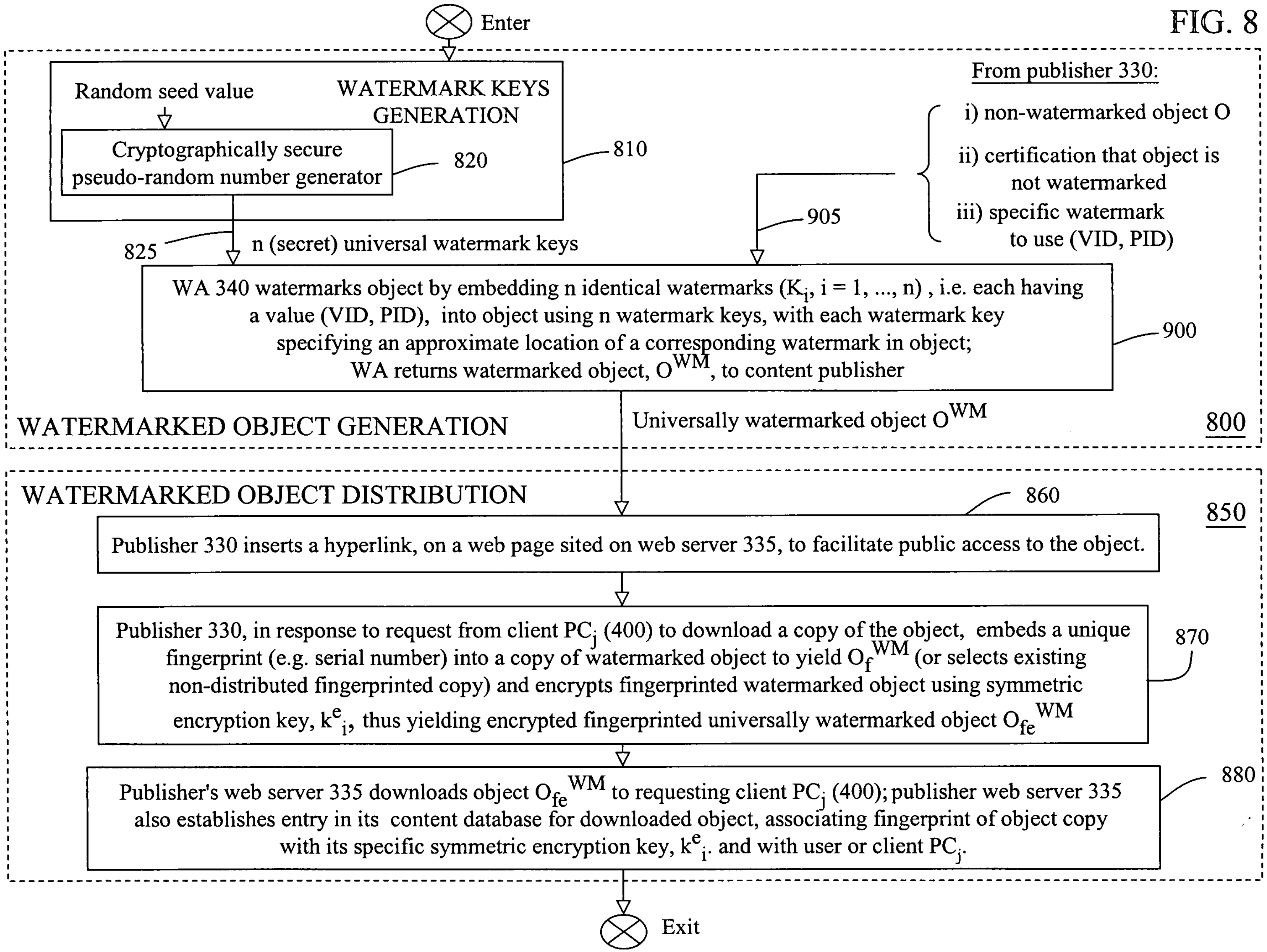

Enter
WATERMARK KEYS GENERATION
Random seed value
Cryptographically secure pseudo-random number generator
820
810
825
n (secret) universal watermark keys
From publisher 330:
i) non-watermarked object O
ii) certification that object is not watermarked
iii) specific watermark to use (VID, PID)
905
WA 340 watermarks object by embedding n identical watermarks (K_i, i = 1, ..., n) , i.e. each having a value (VID, PID), into object using n watermark keys, with each watermark key specifying an approximate location of a corresponding watermark in object; WA returns watermarked object, O^{WM}, to content publisher
900
Universally watermarked object O^{WM}
WATERMARKED OBJECT GENERATION
800
WATERMARKED OBJECT DISTRIBUTION
860
850
Publisher 330 inserts a hyperlink, on a web page sited on web server 335, to facilitate public access to the object.
Publisher 330, in response to request from client PC_j (400) to download a copy of the object, embeds a unique fingerprint (e.g. serial number) into a copy of watermarked object to yield O_f^{WM} (or selects existing non-distributed fingerprinted copy) and encrypts fingerprinted watermarked object using symmetric encryption key, k^e_i, thus yielding encrypted fingerprinted universally watermarked object O_{fe}^{WM}
870
Publisher's web server 335 downloads object O_{fe}^{WM} to requesting client PC_j (400); publisher web server 335 also establishes entry in its content database for downloaded object, associating fingerprint of object copy with its specific symmetric encryption key, k^e_i. and with user or client PC_j.
880
Exit

FIG. 8

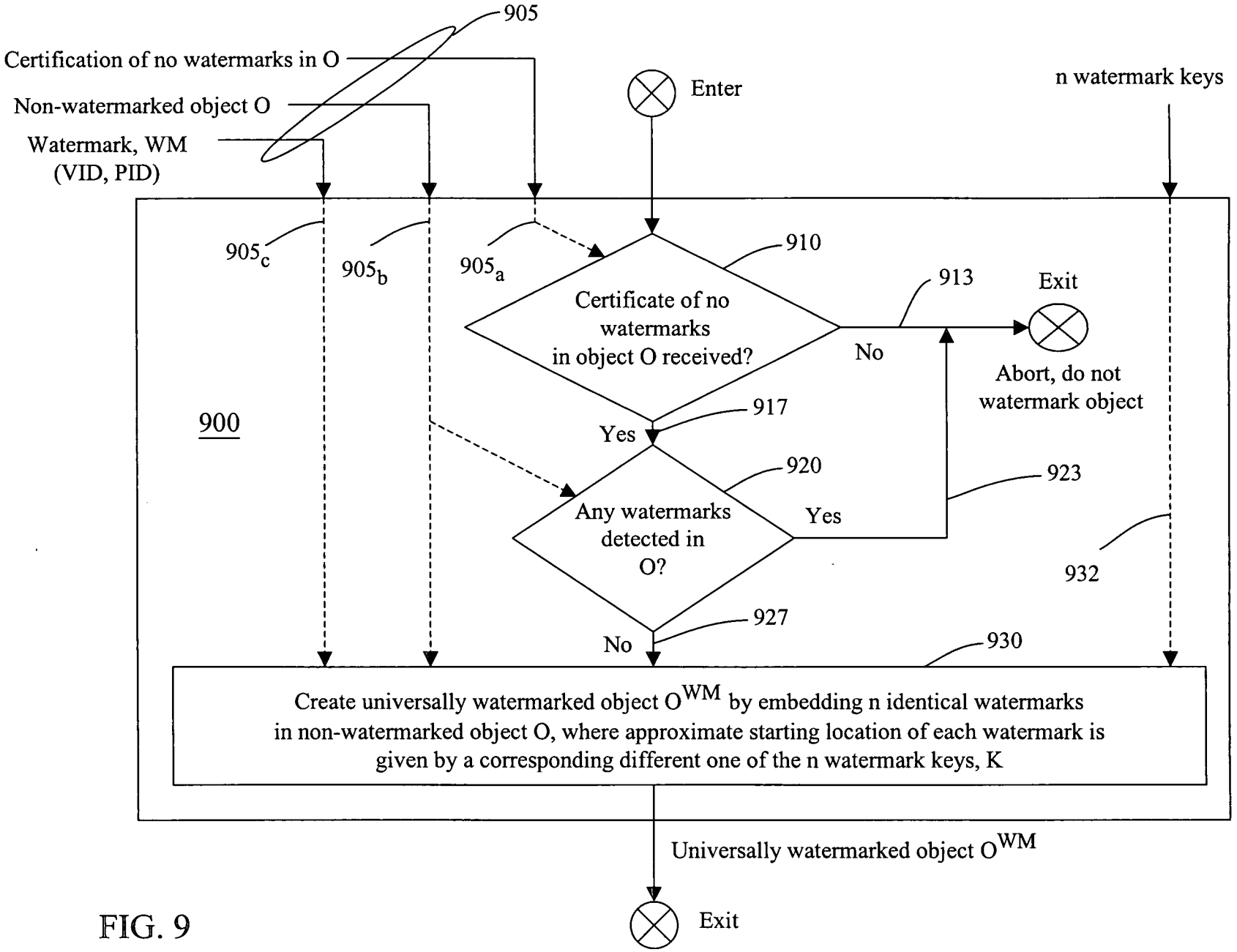

905
Certification of no watermarks in O
Non-watermarked object O
Watermark, WM
(VID, PID)
Enter
n watermark keys
905c
905b
905a
910
Certificate of no watermarks in object O received?
913
Exit
No
Abort, do not watermark object
900
917
Yes
920
923
Any watermarks detected in O?
Yes
932
927
No
930
Create universally watermarked object O^WM by embedding n identical watermarks in non-watermarked object O, where approximate starting location of each watermark is given by a corresponding different one of the n watermark keys, K
Universally watermarked object O^WM
Exit

FIG. 9

FIG. 11

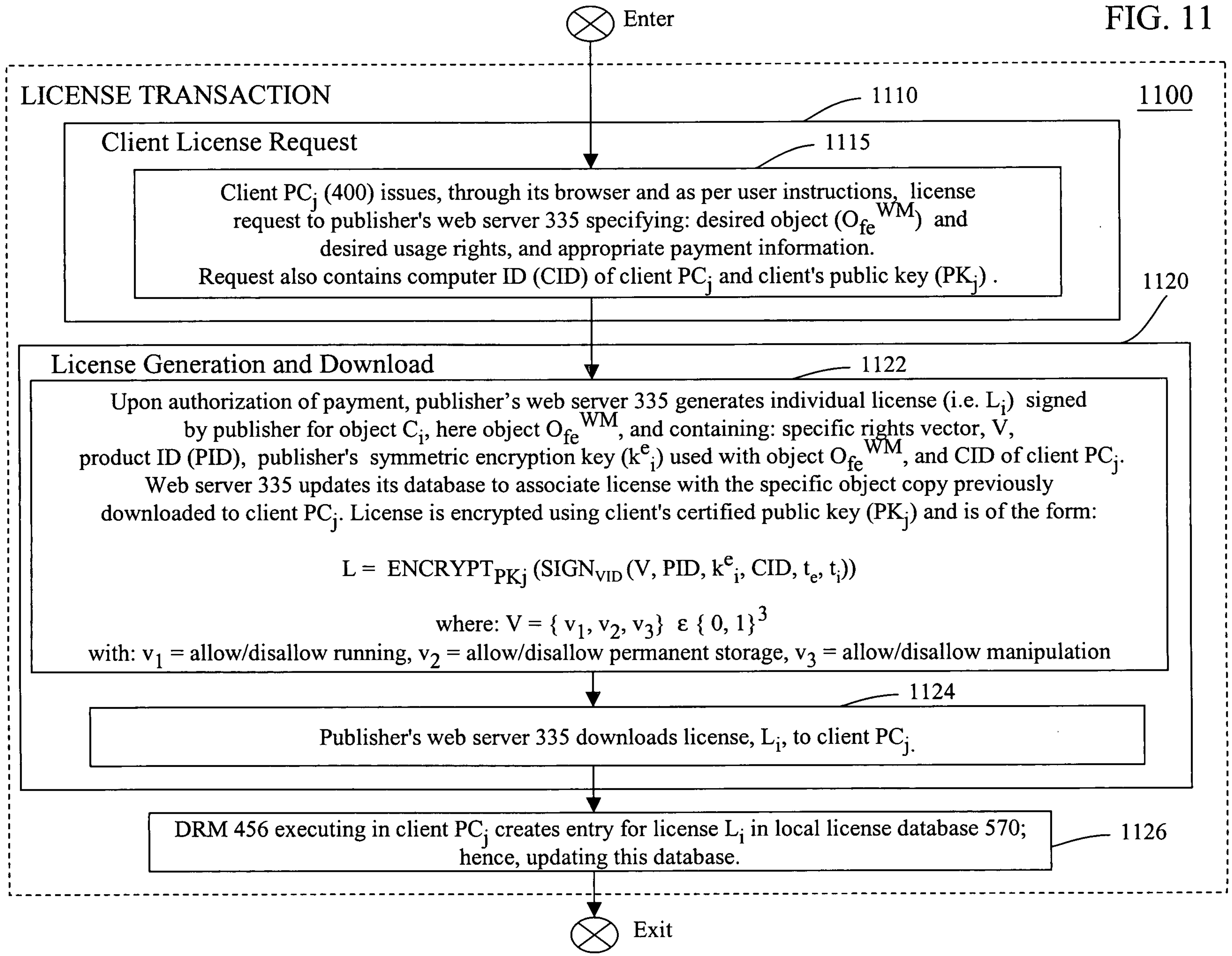

Enter
LICENSE TRANSACTION
1100
1110
Client License Request
1115
Client PC_j (400) issues, through its browser and as per user instructions, license request to publisher's web server 335 specifying: desired object (O_{fe}^{WM}) and desired usage rights, and appropriate payment information.
Request also contains computer ID (CID) of client PC_j and client's public key (PK_j).
1120
License Generation and Download
1122
Upon authorization of payment, publisher's web server 335 generates individual license (i.e. L_i) signed by publisher for object C_i, here object O_{fe}^{WM}, and containing: specific rights vector, V, product ID (PID), publisher's symmetric encryption key (k^e_i) used with object O_{fe}^{WM}, and CID of client PC_j. Web server 335 updates its database to associate license with the specific object copy previously downloaded to client PC_j. License is encrypted using client's certified public key (PK_j) and is of the form:
$L = ENCRYPT_{PKj}(SIGN_{VID}(V, PID, k^e_i, CID, t_e, t_i))$
where: $V = \{ v_1, v_2, v_3 \} \; \varepsilon \; \{0, 1\}^3$
with: v_1 = allow/disallow running, v_2 = allow/disallow permanent storage, v_3 = allow/disallow manipulation
1124
Publisher's web server 335 downloads license, L_i, to client PC_j.
DRM 456 executing in client PC_j creates entry for license L_i in local license database 570; hence, updating this database.
1126
Exit

FIG. 12

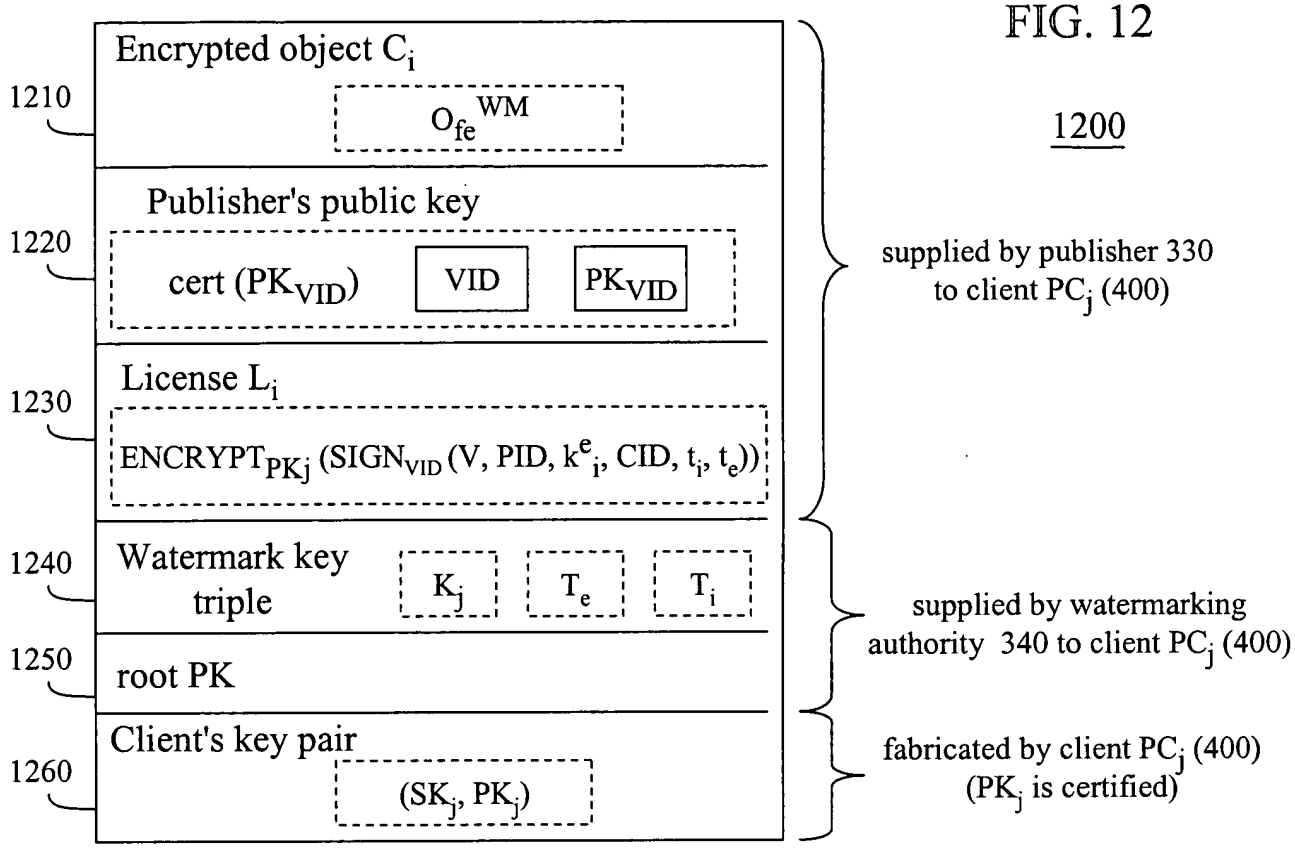

1200
1210
Encrypted object C_i
O_{fe}^{WM}
1220
Publisher's public key
cert (PK_{VID})
VID
PK_{VID}
1230
License L_i
$ENCRYPT_{PKj}$ ($SIGN_{VID}$ (V, PID, k^e_i, CID, t_i, t_e))
supplied by publisher 330 to client PC_j (400)
1240
Watermark key triple
K_j
T_e
T_i
1250
root PK
supplied by watermarking authority 340 to client PC_j (400)
1260
Client's key pair
(SK_j, PK_j)
fabricated by client PC_j (400) (PK_j is certified)

FIG. 14

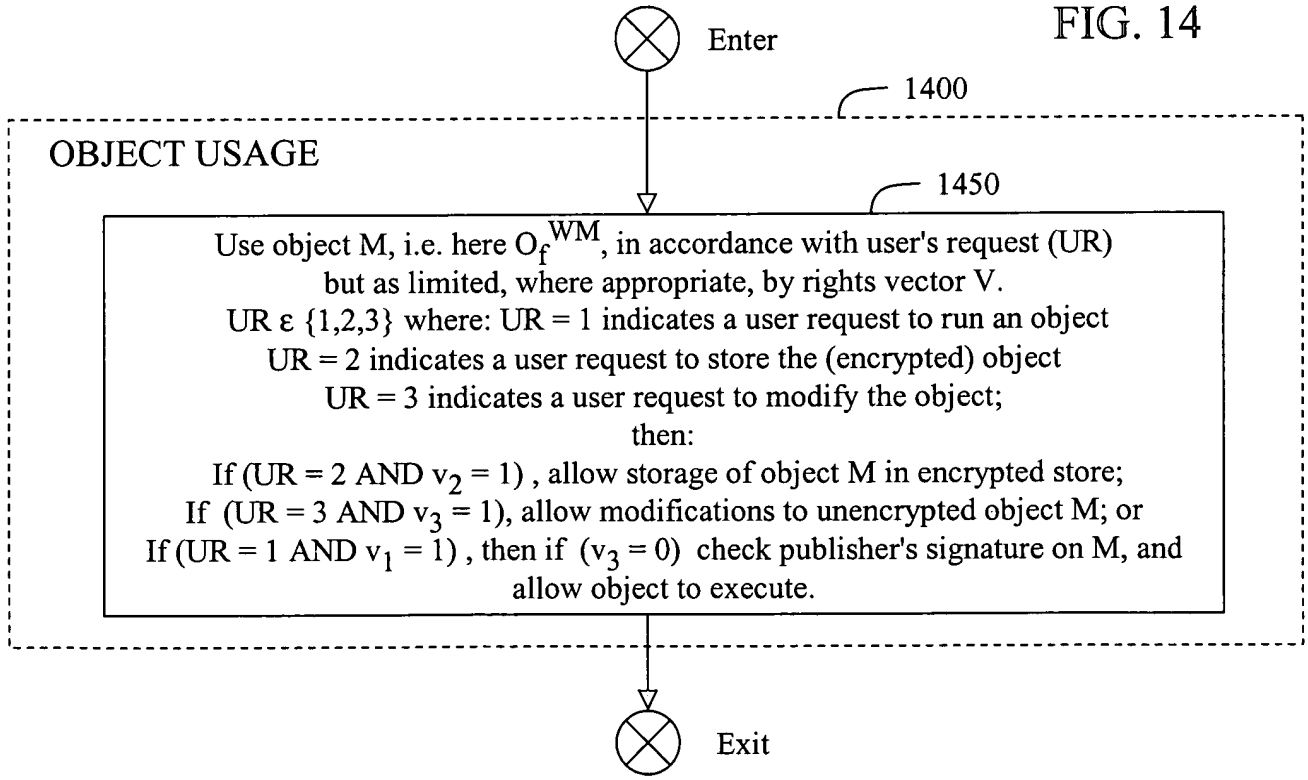
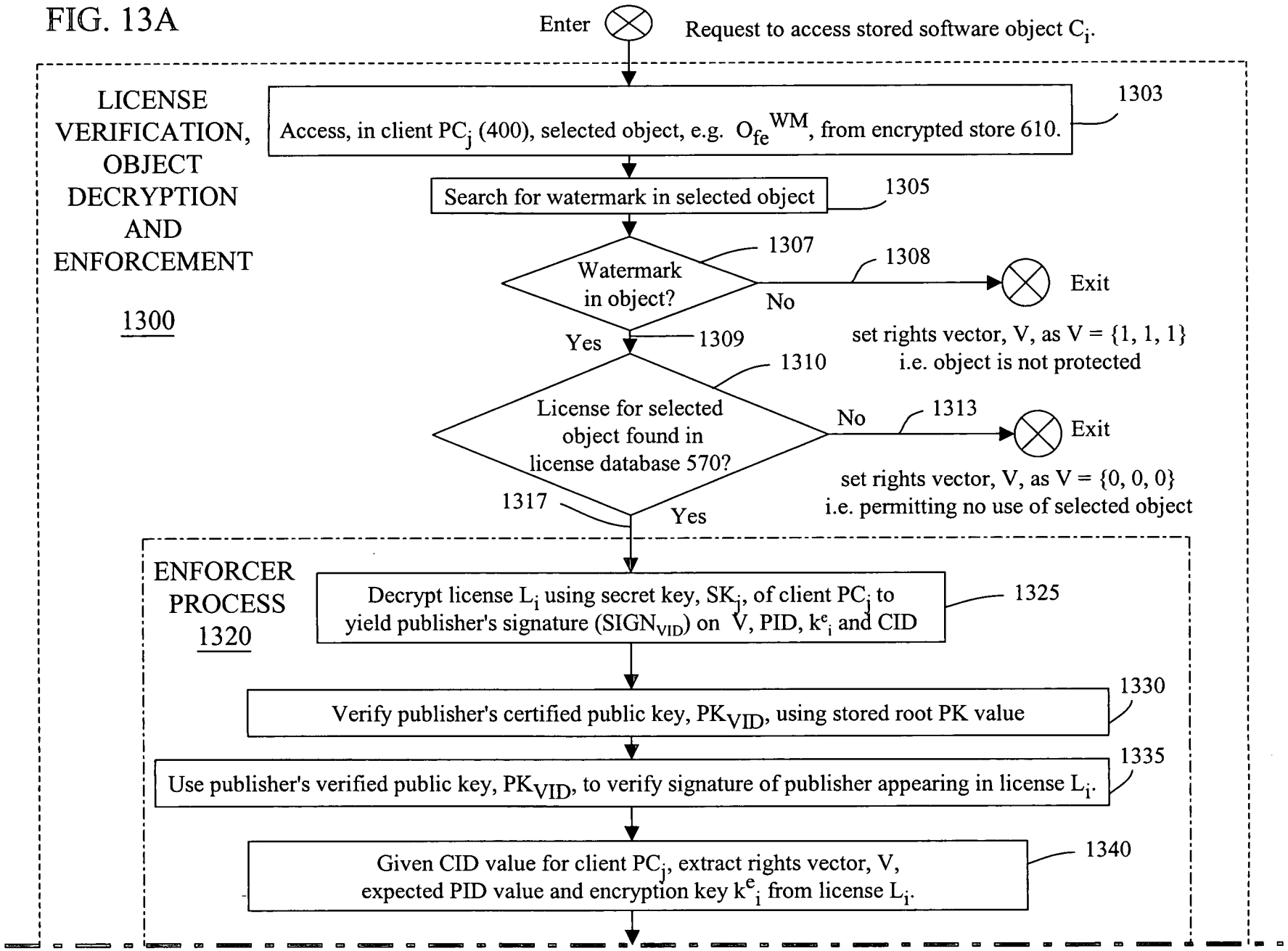

Enter
1400
OBJECT USAGE
1450
Use object M, i.e. here O_f^{WM}, in accordance with user's request (UR)
but as limited, where appropriate, by rights vector V.
UR ε {1,2,3} where: UR = 1 indicates a user request to run an object
UR = 2 indicates a user request to store the (encrypted) object
UR = 3 indicates a user request to modify the object;
then:
If (UR = 2 AND v_2 = 1) , allow storage of object M in encrypted store;
If (UR = 3 AND v_3 = 1), allow modifications to unencrypted object M; or
If (UR = 1 AND v_1 = 1) , then if (v_3 = 0) check publisher's signature on M, and
allow object to execute.
Exit

FIG. 13A


Enter
Request to access stored software object C_i.
LICENSE VERIFICATION, OBJECT DECRYPTION AND ENFORCEMENT
1300
Access, in client PC_j (400), selected object, e.g. O_{fe}^{WM}, from encrypted store 610.
1303
Search for watermark in selected object
1305
Watermark in object?
1307
1308
No
Exit
set rights vector, V, as V = {1, 1, 1}
i.e. object is not protected
Yes
1309
License for selected object found in license database 570?
1310
No
1313
Exit
set rights vector, V, as V = {0, 0, 0}
i.e. permitting no use of selected object
1317
Yes
ENFORCER PROCESS
1320
Decrypt license L_i using secret key, SK_j, of client PC_j to yield publisher's signature ($SIGN_{VID}$) on V, PID, k^e_i and CID
1325
Verify publisher's certified public key, PK_{VID}, using stored root PK value
1330
Use publisher's verified public key, PK_{VID}, to verify signature of publisher appearing in license L_i.
1335
Given CID value for client PC_j, extract rights vector, V, expected PID value and encryption key k^e_i from license L_i.
1340

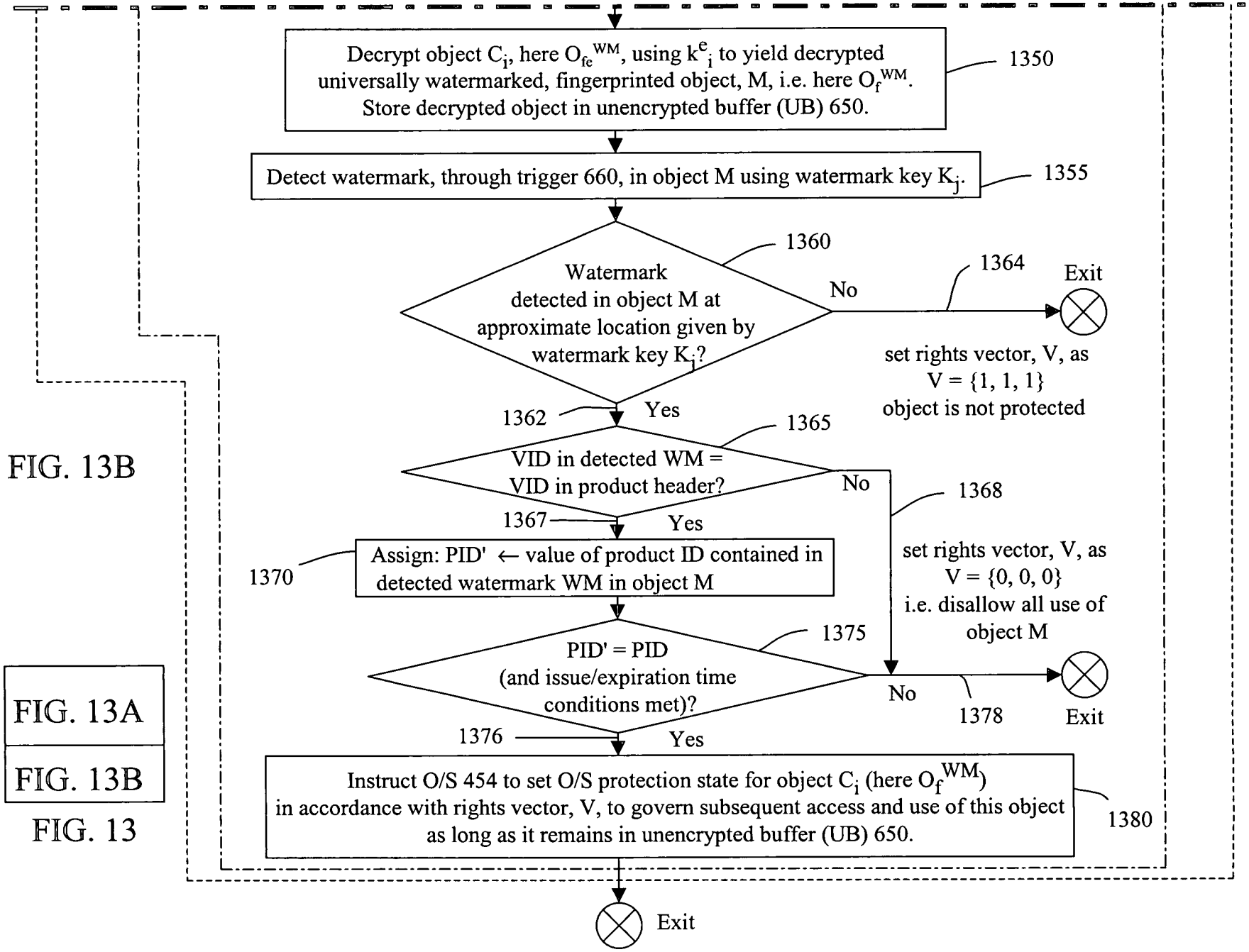


FIG. 13B

FIG. 13A

FIG. 13B

FIG. 13

FIG. 15 -- CLIENT WATERMARK KEY ASSIGNMENT PROCESS

1500

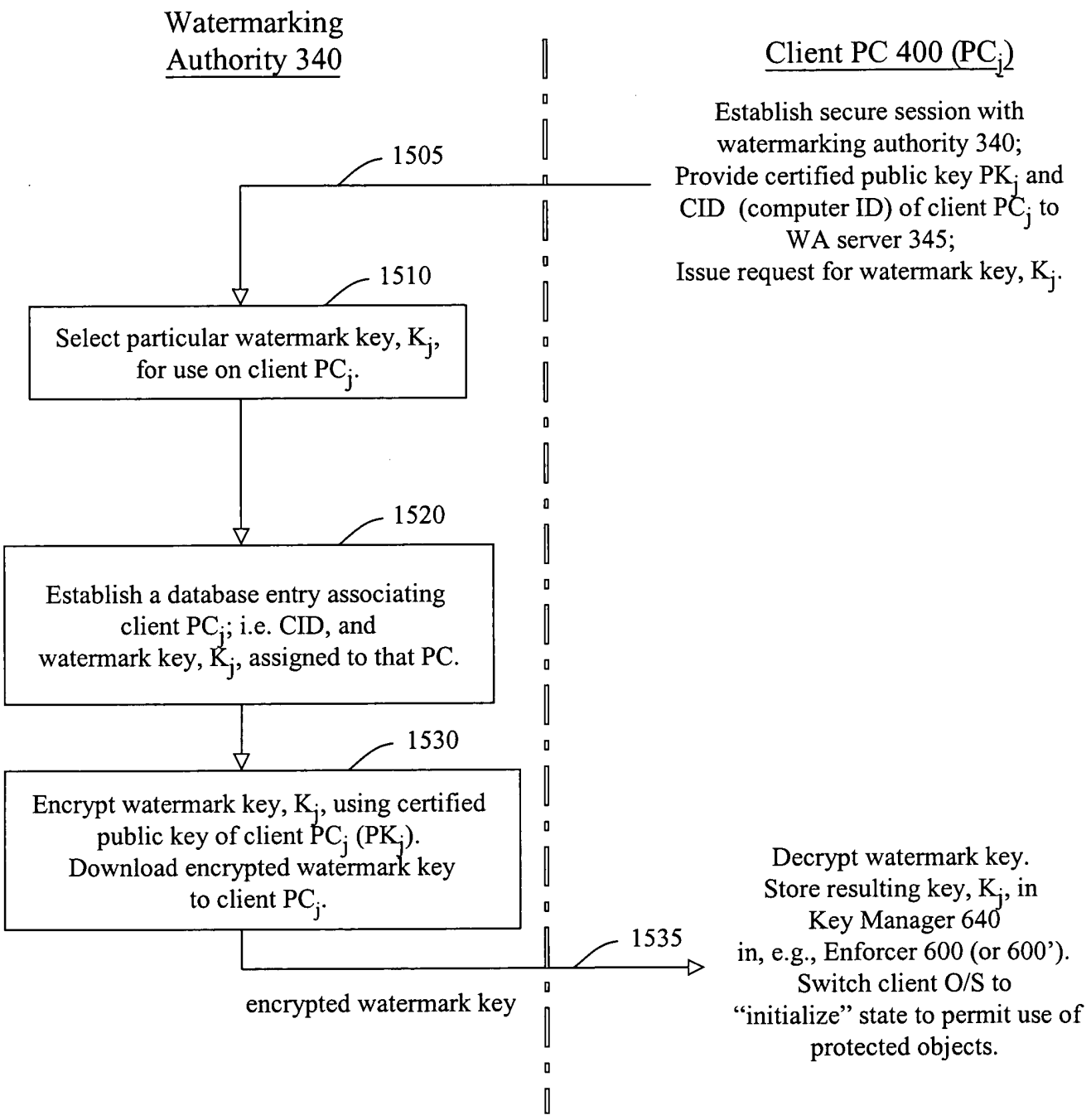


FIG. 16 -- NEW WATERMARK KEY PROVISIONING PROCESS

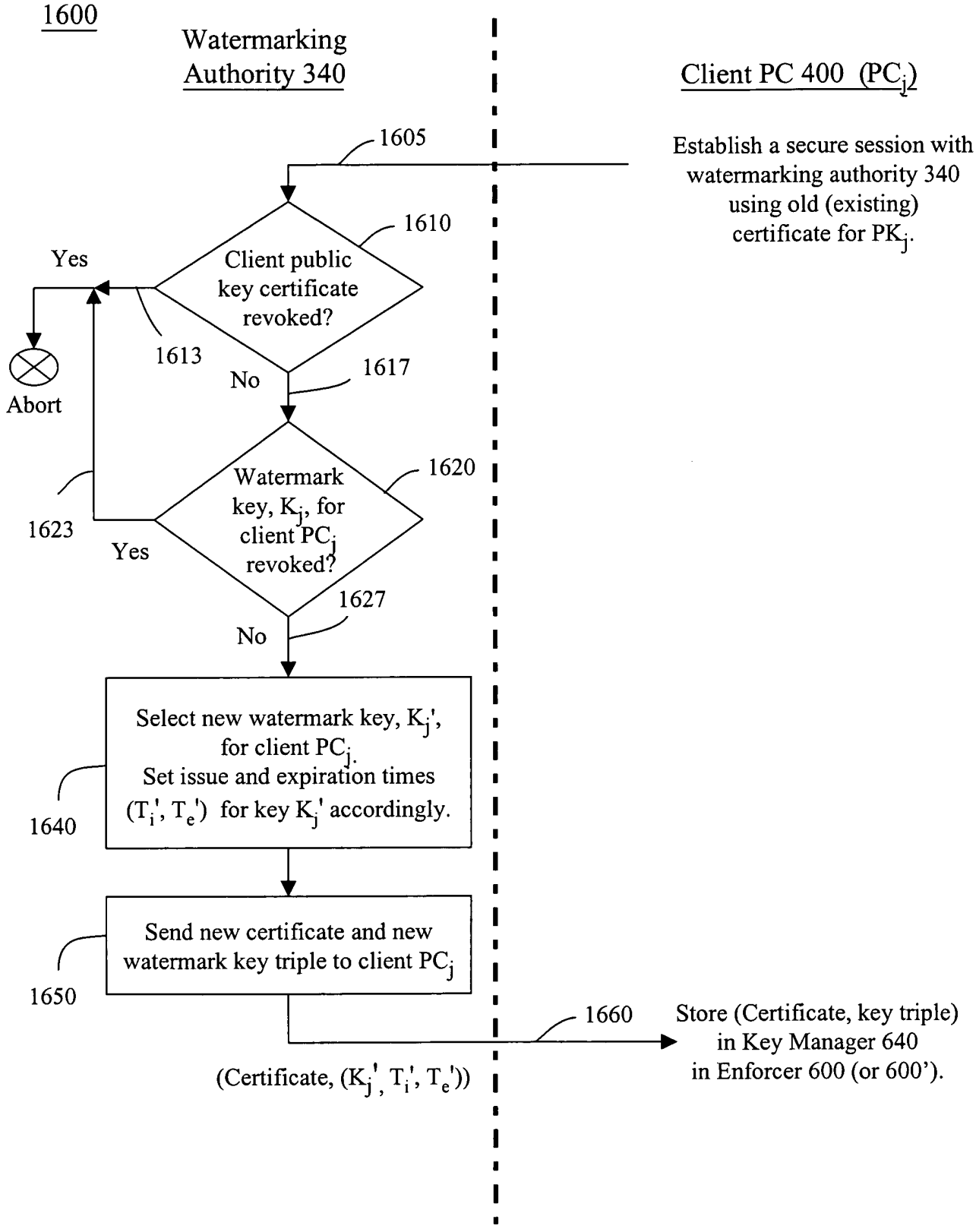
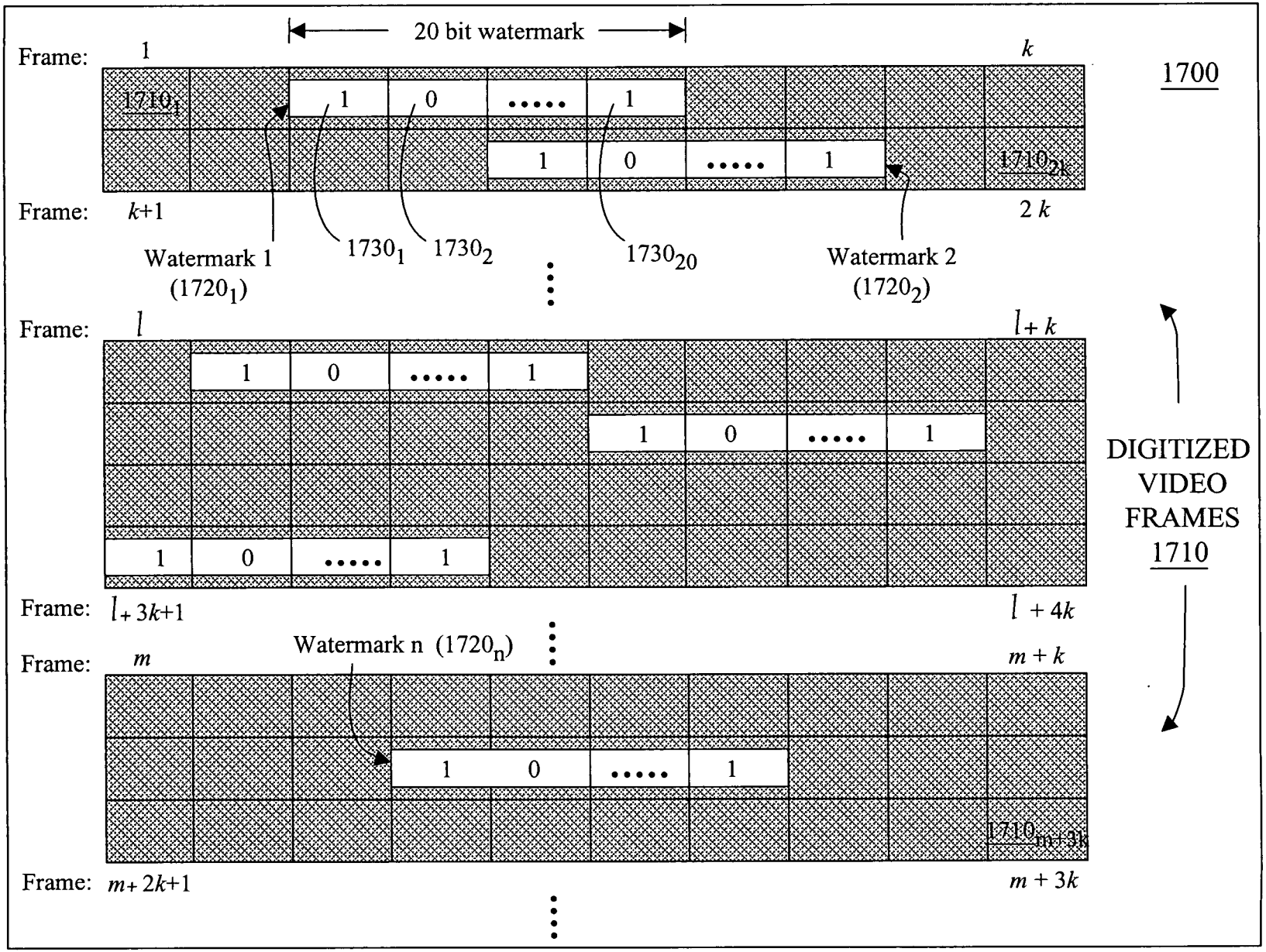


FIG. 17


20 bit watermark
Frame:
1
k
1700
1710$_1$
1
0
.....
1
1
0
.....
1
1710$_{2k}$
Frame:
k+1
2 k
Watermark 1
(1720$_1$)
1730$_1$
1730$_2$
1730$_{20}$
Watermark 2
(1720$_2$)
Frame:
l
l+ k
1
0
.....
1
1
0
.....
1
1
0
.....
1
Frame:
l+ 3k+1
l + 4k
Watermark n (1720$_n$)
Frame:
m
m + k
1
0
.....
1
1710$_{m+3k}$
Frame:
m+ 2k+1
m + 3k
DIGITIZED
VIDEO
FRAMES
1710

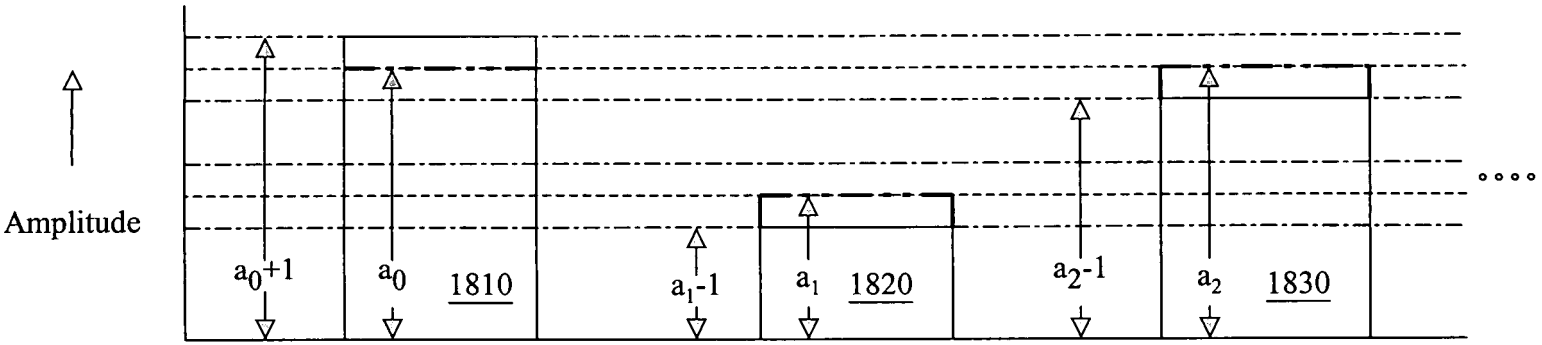

Amplitude
a_0+1
a_0
1810
a_1-1
a_1
1820
a_2-1
a_2
1830
Pixels

FIG. 18

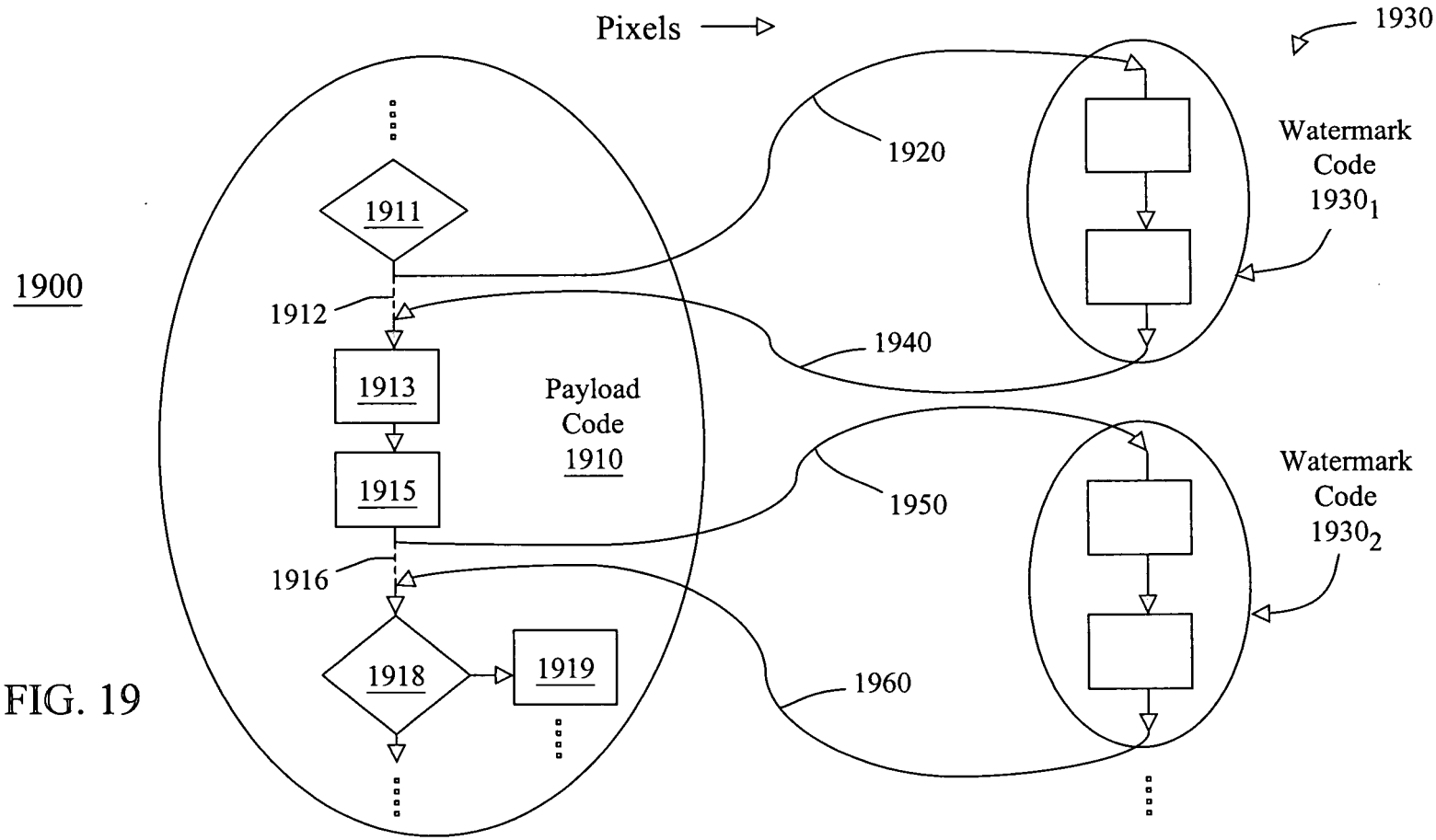

1900
1911
1912
1913
1915
1916
1918
1919
Payload Code 1910
1920
1930
Watermark Code 1930_1
1940
1950
Watermark Code 1930_2
1960

FIG. 19